

Transcrição

Nesta etapa, falaremos sobre as **Medidas de Dispersão**.

Anteriormente, abordamos as Medidas de Tendência Central e Separatrizes que resumizam o dataset analisados.

Porém, nem sempre são suficientes para distinguir conjunto de dados diferentes, principalmente quando têm uma **variação** muito significativa.

Se pegarmos as notas médias do `df` que criamos como exemplo com o boletim de três alunos em sete matérias escolares, veremos que Fulano e Sic

```
df.mean()
```

Matérias

Beltrano 5.142867

Fulano 7.714286

Sicrano 7.714286

dtype: float64

Como temos poucas observações nesse conjunto, poderemos fazer uma análise prévia rapidamente com a tabela:

`df`

Matérias	Beltrano	Fulano	Sicrano
Matemática	10.0	8	7.5
Português	2.0	10	8.0
Inglês	0.5	4	7.0
Geografia	1.0	8	8.0
História	3.0	6	8.0
Física	9.5	10	8.5
Química	10.0	8	7.0

Sicrano possui notas menos dispersas e mais constantes, com notas altas e valores próximos em todas as matérias. Ao passo que Fulano apresenta un

Se calcularmos a mediana, veremos que ambos os alunos possuem o mesmo resultado também.

```
df.median()
```

Matérias

Beltrano 3.0

Fulano 8.0

```
Sicrano      8.0
```

```
dtype: float64
```

Ou seja, essas estatísticas de tendência central que estudamos não identificam essas questões que observamos diretamente no boletim. Porém, com u

Portanto, precisaremos da informação de Dispersão.

Começaremos abordando o **desvio médio absoluto** com a seguinte fórmula:

$$DM = \frac{1}{n} \sum_{i=1}^n |X_i - \bar{X}|$$

"DM" é igual ao somatório desses desvios, formado pelo módulo de "X" índice "i" que é o valor de cada nota do `df` menos a média geral "X". A últ

positivos.

Em um caso onde a média é maior do que o valor, como o complemento de "X" sendo igual a "10" e "X" no índice "i" for igual a "2" por exemplo, o

subtração, e entenderemos o porquê adiante.

Aplicaremos esta estatística para Fulano na célula nesta parte do notebook. Chamaremos a variável como `notas_fulano` sendo igual a `df[]` conte

```
notas_fulano = df['Fulano']
notas_fulano
```

```
Matemática      8
Português       10
Inglês          4
Geografia       8
História        6
Física          10
Química         8
Name: Fulano, dtype: int64
```

Se aplicarmos ainda mais um par de colchetes dentro de `df[]` , criaremos um `DataFrame` de fato com Pandas.

```
notas_fulano = df[['Fulano']]
notas_fulano
```

Matérias	Fulano
Matemática	8
Português	10
Inglês	4
Geografia	8
História	6
Física	10
Química	8

Precisaremos de um `DataFrame` para adicionarmos novas variáveis que nos ajudarão a entendermos os cálculos melhor.

Agora que já temos o "X" índice "i", descobriremos a média com uma nova variável `nota_media_fulano` sendo igual a `notas_fulano` com `.mean`

```
nota_media_fulano = notas_fulano.mean()[0]
nota_media_fulano
```

Como resultado, obteremos a média `7.714285714285714`.

Em seguida, criaremos o `'Desvio'` dentro do `notas_fulano[]`. este será igual a `notas_fulano[]` recebendo `'Fulano'` menos `nota_media_fulano`

```
notas_fulano['Desvio'] = notas_fulano['Fulano'] - nota_media_fulano
notas_fulano
```

Matérias	Fulano	Desvio
Matemática	8	0.285714
Português	10	2.285714
Inglês	4	-3.714286
Geografia	8	0.285714
História	6	-1.714286
Física	10	2.285714
Química	8	0.285714

Notaremos a presença de alguns valores negativos nos casos em que a nota é menor do que a média.

Porém, se pegarmos `notas_fulano[]` com o `'Desvio'` e somarmos com `.sum()` para aplicarmos a fórmula, teremos um valor negativo muito pr

```
notas_fulano['Desvio'].sum()
```

```
-8.881784197001252e-16
```

Isso acontece por conta das casas decimais, mas consideraremos zero absoluto.

Logo, se fizermos a conta de "0" dividido por "n", o resultado será "0" e o desvio médio não fará sentido. Portanto, pegaremos apenas os valores pos

Para isso, colocaremos `Desvio` entre barras e aspas simples sendo igual ao desvio médio com o método `.abs()`, o qual pegará um valor absoluto.

```
notas_fulano['|Desvio|'] = notas_fulano['Desvio'].abs()
notas_fulano
```

Matérias	Fulano	Desvio	/Desvio/
Matemática	8	0.285714	0.285714
Português	10	2.285714	2.285714
Inglês	4	-3.714286	3.714286
Geografia	8	0.285714	0.285714
História	6	-1.714286	1.714286
Física	10	2.285714	2.285714
Química	8	0.285714	0.285714

Com isso, veremos os valores todos positivos que desconsideram sinais, e poderemos fazer a somatória em questão.

Em seguida, criaremos um gráfico de demonstração da variável `ax` com diversas configurações que podem ser observadas com calma posteriormen

```
ax = notas_fulano['Fulano'].plot(style = 'o')
ax.figure.set_size_inches(14, 6)
ax.hlines(y = nota_media_fulano, xmin = 0, xmax = notas_fulano.shape[0] - 1, colors = 'red')
for i in range(notas_fulano.shape[0]):
    ax.vlines(x = i, ymin = nota_media_fulano, ymax = notas_fulano['Fulano'][i], linestyle='dashed')
ax
```



A reta central representa a média de nossos dados, cada marcação com *bolinhas* serão os dados observados, e as linhas tracejadas serão os desvios.

No caso do desvio médio, esses valores negativos se tornarão positivos, pois os estamos pegando sem o sinal.

Em seguida, calcularemos a média das sete marcações para a Medida de Dispersão e variação dos dados.

Na célula seguinte, faremos o cálculo de fato; pegaremos a média `.mean()` de `notas_fulano[]` com `'|Desvio|'`.

```
notas_fulano['|Desvio|'].mean()
```

Como resultado, obteremos o valor `1.5510204081632648` obtido pela fórmula da média dos desvios.

Com isso, chegaremos a conclusão de que o Desvio Médio Absoluto representado pela variável `desvio_medio_absoluto` será igual a `notas_fulano` calculará este valor, e então exibiremos o resultado.

```
desvio_medio_absoluto = notas_fulano['Fulano'].mad()  
desvio_medio_absoluto
```

O retorno apresentará o mesmo valor do comando anterior, portanto utilizar esta função é bem mais eficiente na área de Ciência ou Estatística de Dados.

A seguir, falaremos sobre as Medidas de Dispersão chamadas **variância** e **desvio padrão**.